## Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (currently amended) A multi-streaming processor ~~system~~ comprising:

a plurality of ~~physical~~ hardware streams for streaming one or more instruction threads;

a set of functional resources coupled to said hardware streams for processing instructions from said streams; ~~and~~

interrupt detection logic, coupled to interrupt signals, for providing said interrupt signals to the processor;

interrupt mapping logic, for providing configurable interrupt mapping of said interrupt signals to ones of said plurality of streams; and

interrupt logic, coupled to said interrupt mapping logic, for interrupting one or more of said streams according to said configurable interrupt mapping provided by said interrupt mapping logic;~~;~~

wherein through ~~the~~ said interrupt logic, ~~specific~~ said interrupts ~~or exceptions~~ are detected, and at the time of their detection ~~a specific~~said ones of said ~~stream of the~~ plurality of ~~physical hardware~~ streams ~~is~~ are directed to process ~~the specific~~ said interrupts ~~or exception~~.

Claim 2 (currently amended) The ~~system~~ processor of claim 1 wherein one of said plurality of interrupts ~~or exception~~ may be mapped to two or more of said plurality of hardware streams.

Claim 3 (currently amended) The ~~system~~ processor of claim 1 wherein ~~two or more interrupts or exceptions may be mapped to one stream~~said plurality of interrupts comprise:

internally generated interrupts;

externally generated interrupts; and

exceptions.~~:~~

Claim 4 (currently amended) The ~~system~~ processor of claim 1 wherein said plurality of interrupts comprise:~~mapping of interrupts to streams is static and determined at processor design.~~

external interrupts generated by a device external to the processor;

internal interrupts intentionally generated by special instructions executed by the processor; and

exceptions caused by execution of an instruction or a hardware error.

Claim 5 (currently amended) The ~~system~~ processor of claim 1 wherein mapping of ~~interrupts and exceptions~~said interrupt signals to ones of said plurality of streams is programmable.

Claim 6 (currently amended) The ~~system~~ processor of claim 5 wherein said configurable interrupt mapping is programmed in a data store, and ~~the~~ said interrupt logic ~~refers~~ references ~~to the~~said data store for mapping ~~data~~ said interrupt signals to ones of said plurality of streams~~to relate received interrupts or exceptions to streams~~.

Claim 7 (canceled)

Claim 8 (canceled)

Claim 9 (currently amended) The ~~system~~ processor of claim 1 wherein ~~the~~ said interrupt signals~~interrupts~~ are software interrupts generated by ~~active~~ said hardware streams.

Claim 10 (currently amended) The ~~system~~ processor of claim 6 wherein ~~the~~ said data store~~storage~~ further comprises a mask for enabling and /disabling execution of ~~mapped interrupts or exceptions~~said interrupt signals.

Claim 11 (currently amended) The ~~system~~ processor of claim 1 wherein, after said interrupt mapping is determined for a detected one of said interrupt ~~or exception~~signals, ~~the~~ said one or more hardware streams are interrupted by ~~the~~ said interrupt logic.

Claim 12 (currently amended) The ~~system~~ processor of claim 11 wherein said one ore more hardware streams interrupted by said interrupt logic~~an interrupted stream~~

acknowledges ~~the~~ said interrupt, and ~~is~~ are vectored to a service ~~routing~~ routine by ~~the~~ said interrupt logic.

Claim 13 (currently amended) The ~~system~~ processor of claim 12 wherein two or more of said hardware streams are interrupted by one of said interrupt ~~or exception~~signals, and wherein ~~the~~ said interrupt logic delays vectoring any of said hardware stream to ~~a~~ said service routine until all interrupted said hardware streams acknowledge ~~the~~ said interrupt signal.

Claim 14 (currently amended) The ~~system~~ processor of claim 13 wherein two of said streams acknowledging ~~the~~ said ~~same~~ interrupt signal are vectored to different service routines by ~~the~~ said interrupt logic.

Claim 15 (currently amended) A method for processing ~~interrupts~~ an interrupt in a multi-stream processor having a plurality of ~~physical~~ hardware streams, comprising ~~steps of~~:

~~(a)~~ detecting ~~an~~ the interrupt ~~or exception~~ and passing the detected interrupt ~~or exception~~ to interrupt mapping logic;

determining, using the interrupt mapping logic, which ones of the plurality of hardware streams are to be interrupted by the interrupt; and

~~(b) directing one or more specific streams of the multi-stream processor to process a specific interrupt or exception at the time of their detection.~~interrupting the ones of the plurality of hardware streams that are to be interrupted by the interrupt.

Claim 16 (currently amended) The method of claim 15 wherein~~, in step (b),~~ the interrupt ~~or exception may be~~is mapped to two or more of the hardware streams.

Claim 17 (canceled)

Claim 18 (currently amended) The method of claim 15 further comprising: ~~wherein~~

mapping of the interrupts to one or more hardware streams ~~is static and determined at processor design~~statically at time of manufacture of the processor.

Claim 19 (currently amended) The method of claim 15 wherein mapping of the interrupts ~~and exceptions~~ is programmable within the processor.

Claim 20 (currently amended) The method of claim 19 wherein the mapping is programmed in a data store, and the interrupt mapping logic refers to the data store for mapping ~~data to relate received~~the interrupt~~s or exceptions~~ to the ones of the plurality of hardware streams.

Claim 21 (canceled)

Claim 22 (currently amended) The method of claim 15 wherein the interrupt~~s are~~ is an external interrupts generated by a devices external to the processor.

Claim 23 (currently amended) The method of claim 15 wherein the interrupt~~s are~~ is a software interrupts generated by one of the plurality of hardware~~active~~ streams.

Claim 24 (currently amended) The method of claim 20 wherein the data storeage ~~further~~ comprises a mask for enabling/disabling execution of the mapped interrupt~~s or exceptions~~.

Claim 25 (canceled)

Claim 26 (currently amended) The method of claim ~~25~~ 15 further comprising:

~~a further step for~~ vectoring ~~an interrupt stream~~the ones of the hardware streams to ~~a~~ service routine~~s~~ after the ~~interrupted~~ hardware streams acknowledges the interrupt.

Claim 27 (currently amended) The method of claim 26 wherein ~~two or more~~the ones of the hardware streams are interrupted by ~~one~~ the interrupt ~~or exception~~, and wherein ~~the~~ interrupt logic delays vectoring any of the hardware streams to a service routine until all of the ones of the hardware~~interrupted~~ streams that are mapped to the interrupt acknowledge the interrupt.

Claim 28 (currently amended) The method of claim 27 wherein ~~two~~ the ones of the hardware streams acknowledging the ~~same~~ interrupt are vectored to different service routines by the interrupt logic.

Claims 29-42 (canceled)